Interface Specification Between
Concrete Batching/Dispatching Software
and
Quality Control/Mix Design Software

Contributors

Adrian Field
Stonemont Solutions

Christopher Kranich
Kronware
Appendix 1

Craig Hanson
GivenHansco
Appendix 2

Eric Palmitesta
MPAQ Automation
Appendix 3

Versions

Version 1 – March 2014
Version 2 – October 2015
- Added Projects
- Added Proposed Methods Section
- Added plant_name to plant list
- Added plant_maxloadsize to plant list
- Documented date and time formats

Version 3 – January 2016
- Added Method Call CustomersSince/{date}
- Added Method Call ProjectsSince/{date}
- Added Batch Results Condensed and method calls.
- Corrected tag <plantlist> to <plants> in Plants schema
- Added Plant List
- Added optional tag mc_plantcode to material list
- Added optional tag mix_plantcode to mix design list
- Added mat_unit, mat_dosagerate to materials
- Added plant_maxloadsizeunit to plants
- Added plant_aggscalecapacity to plants
- Added plant_cemscalecapacity to plants
- Added plant_scalecapacityunit to plants

Version 3.1 – January 2017

- Added Method Call CustomerContainsList/{search}

Version 3.2 – February 2017

- Added mix_slumpmin, mix_slumpmax to mix designs
- Added mat_solidspct to materials
- Added mc_solidspct to mix design components
- Added bt_custname and bt_jobname to batch results
- Added support for additional batchresultlist methods for filtering by customer and job.

Purpose

Concrete batching/dispatching and quality control/mix design software have a mutually beneficial need to communicate with each other. Currently, no universal application programming interface (API) exists for this purpose other than the ulink protocol, which is seriously outdated and limited. The goal of this document is to outline an API specification that can be used between concrete batching/dispatching software and quality control/mix design software. It also could be expanded to be used for other purposes including the communication between central dispatch systems and local batch systems. However, the goal is that the specification be maintained for the benefit of all companies rather than individual companies modifying the specification at will.

This API uses the extensible markup language (XML) or the JavaScript Object Notation (JSON). XML is a text-based open standard markup language for encoding documents in a format that is readable by both machines and humans. JSON is a lightweight text-based open standard that is readable by both machines and humans. The XML and JSON formats were chosen for this API as they are widely used and they are simple to read, write, understand, use, and extend. Each batching/dispatching software company can decide independently which format to use. The name of this API is BCQCI.

The method of communication between the batching/dispatching and quality control/mix design software will depend on the capabilities of each system. The preferred approach is to use a RESTful web service, which is a method of communicating over the internet or intranet. REST is an acronym for representational state transfer. The RESTful web service approach can provide a very robust interface between the software products since data can be both requested and submitted in real-time. The web service approach does not require specific knowledge of the other system to implement the interface. An alternative approach is to use file transfers where each system monitors folders and imports/exports the files. The file transfer approach, although limited, can still provide an alternative to the ulink protocol.

The current implementation of the BCQCI allows for transfer of information related to the following: Materials, Mix Designs, Batch Results, Customers, and Plants. Materials, such as aggregates, are commonly tested and managed using quality control software. Mix designs are commonly developed and tested using specialized software that includes both mix design and quality control functionality. Test results, such as gradations, are commonly used in the development and management of mix designs but are rarely managed by batching software. Batch results are created when mix designs are batched by batching software but are very usefulness from a quality control stand-point. Customers and plants are features commonly shared by both batching and quality control software. Since the information related to concrete mix design development, testing, and batching is commonly performed by different software packages it is advantageous to have a common interface and seamless integration between the different software programs that use this information. Hence, the development of BCQCI to meet the need for a current universal API between concrete batching/dispatching and quality control/mix design software.

XML with Definitions

Note that BCQCI is case-sensitive so all XML and JSON tags are required to be in lower case. Strings that could be interpreted as markup characters need to be HTML encoded.

The header tag is included in all xml/json definitions. It is the first tag following the highest level document tag, which is <bcqci>.

Header

```
<request_status>
  <status /> (Required)
        <!—return status (Enumeration)-->
  <msg /> (Optional)
        <!—return message.  When status is not ok, this is an error message (Alphanumeric)-->
   <culture /> (Optional)
        <!--culture code (Enumeration)-->
        <!--If not provided en should be assumed-->
 </request_status>
```

Mix Designs

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <mixdesigns>
 <mixdesign>
 <mixinfo>
  <mix_plantcode /> (Required)
        <!--user-defined code that uniquely identifies plant (Alphanumeric)-->
  <mix_code /> (Required)
        <!--user-defined code that uniquely identifies mix at this plant (Alphanumeric)-->
  <mix_name /> (Optional)
        <!--user-defined mix name (Alphanumeric)-->
  <mix_unittype /> (Optional)
        <!--master mix unit type (Enumeration)-->
        <!--If not provided US should be assumed-->
  <mix_ssd /> (Optional)
        <!--Aggregate weight and specific gravity type (Enumeration)-->
        <!--If not provided ssd should be assumed for aggregates-->
  <mix_maxsize /> (Optional)
        <!--Maximum load size (Numeric)-->
        <!--Typically not provided by quality systems-->
  <mix_maxsizeunit /> (Optional)
        <!--Maximum load size unit (Enumeration)-->
        <!--If not provided defaults are US=yd3, Metric=m3-->
  <mix_mixtime /> (Optional)
        <!--Mixing time (Numeric)-->
```

<!--Typically not provided by quality systems-->
<mix_mixtimeunit /> (Optional)
<!--Mixing time unit (Enumeration)-->
<!--If not provided default is seconds->
<mix_price /> (Optional)
<!--Mix price (Numeric)-->
<mix_cstrength /> (Optional)
<!--Mix design compressive Strength (Numeric)-->
<mix_cstrengthunit /> (Optional)
<!--Mix design compressive strength unit (Enumeration)-->
<!--If not provided defaults are US=psi, Metric=MPa-->
<mix_slump /> (Optional)
<!--Mix design slump (Numeric)-->
<mix_slumpmin /> (Optional)
<!--Mix design minimum slump (Numeric)-->
<mix_slumpmax /> (Optional)
<!--Mix design maximum slump (Numeric)-->
<mix_slumpunit /> (Optional)
<!--Mix design slump unit (Enumeration)-->
<!--if not provided defaults are US=in, Metric=mm-->
<mix_air /> (Optional)
<!--Mix design air content % (Numeric)-->
<mix_wcmratio /> (Optional)
<!--Mix design water/cementitious ratio (Numeric)-->
<mix_aggsize /> (Optional)
<!--Mix design aggregate maximum nominal size (Numeric)-->
<mix_aggsizeunit /> (Optional)
<!--mix design aggregate maximum nominal size unit (Enumeration)-->
<!--If not provided defaults are US=in, Metric=mm-->
<mix_yield /> (Optional)
<!--mix design yield (Numeric)-->
<mix_yieldunit /> (Optional)
<!--mix design yield unit (Enumeration)-->
<!--If not provided defaults are US=yd3, Metric=m3-->
<mix_notes /> (Optional)
<!--mix notes (Alphanumeric)-->
<mix_udf1 /> (Optional)
<!--mix user-defined field. (Alphanumeric)-->
<mix_udf2 /> (Optional)
<!--mix user-defined field. (Alphanumeric)-->
<mix_udf3 /> (Optional)
<!--mix user-defined field. (Alphanumeric)-->
</mixinfo>
<mixcomponents>
<mixcomponent>
<mc_mixcode /> (Required)

```
        <!--user-defined mix code that matches mix_code (Alphanumeric)-->
  <mc_code /> (Required)
        <!--user-defined code that uniquely identifies this component in the mix (Alphanumeric)-->
        <!--should match a material code at plant or one is created-->
  <mc_name />  (Optional)
        <!--user-defined component name (Alphanumeric)-->
  <mc_type /> (Required)
        <!--Material type (Enumeration)-->
  <mc_quantity /> (Required)
        <!--Material quantity (Numeric)-->
  <mc_unit /> (Required)
         <!--Material quantity unit (Enumeration)  -->
         <!--Defaults for US unit types: fl oz for admixtures, gallons for water, pounds for
         aggregates, cements.-->
         <!--Defaults for Metric unit types: ml for admixtures, liters for water, kilograms for
         aggregates, cements.-->
  <mc_dosagetype /> (Optional)
        <!--Material dosage type (Enumeration)-->
        <!--If not provided default is per yd3/m3-->
  <mc_solidspct /> (Optional)
        <!—Admixture % Solids (Numeric)-->
  <mc_order /> (Optional)
        <!--Material load-out order or rank (Integer)-->
        <!--typically not included from quality systems-->
  <mc_source /> (Optional)
        <!--Material source (Alphanumeric)-->
        <!--included in case material needs to be created at plant or is stored with component-->
  <mc_spgr /> (Optional)
        <!--Material specific gravity (Numeric)-->
        <!--included in case material needs to be created at plant or is stored with component→
        <!--If not provided default is 1-->
  <mc_absorption /> (Optional)
        <!--Material absorption % (Numeric)-->
        <!--included in case material needs to be created at plant or is stored with component→
        <!--If not provided default is 0-->
 </mixcomponent>
 </mixcomponents>
 </mixdesign>
 </mixdesigns>
</bcqci>
```

Materials

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<bcqci>
 <materials>
 <material>
  <mat_plantcode /> (Required)
        <!--User-defined code that uniquely identifies this plant (Alphanumeric)-->
  <mat_code /> (Required)
        <!--User-defined code that uniquely identifies this material at this plant  (Alphanumeric)-->
  <mat_name /> (Optional)
        <!--Material name  (Alphanumeric)-->
  <mat_source /> (Optional)
        <!--Material source  (Alphanumeric)-->
  <mat_type /> (Required)
        <!--Material type (Enumeration)-->
  <mat_unit /> (Optional)
         <!--Material default batch unit (Enumeration)  -->
         <!--Defaults for US unit types: fl oz for admixtures, gallons for water, pounds for
         aggregates, cements.-->
         <!--Defaults for Metric unit types: ml for admixtures, liters for water, kilograms for
         aggregates, cements.-->
  <mat_dosagetype /> (Optional, applies to admixtures only)
        <!--Material dosage type (Enumeration)-->
        <!--If not provided default is per yd3/m3-->
  <mat_dosagerate /> (Optional)
        <!—Admixture default dosage quantity (Numeric)-->
  <mat_solidspct /> (Optional)
        <!—Admixture % Solids (Numeric)-->
  <mat_order /> (Optional)
        <!--Material load-out order or rank (Integer)-->
        <!--typically not included from quality systems→
  <mat_price /> (Optional)
        <!--Material price  (Numeric)-->
  <mat_ssd /> (Optional)
        <!--Aggregate specific gravity type (Enumeration)-->
        <!--If not provided ssd should be assumed for aggregates-->
  <mat_spgr /> (Required for Aggregate and Cement)
        <!--Material specific gravity  (Numeric)-->
        <!--If not provided default is 1-->
  <mat_absorption /> (Optional, applies to aggregates only)
        <!--Material absorption %  (Numeric)-->
        <!--If not provided default is 0-->
  <mat_totalmoisture /> (Optional, applies to aggregates only)
        <!--Material total moisture %  (Numeric)-->
        <!--If not provided default is 0-->
  <mat_udf1 /> (Optional)
        <!--material user-defined field. (Alphanumeric)-->
```

```
  <mat_udf2 /> (Optional)
        <!--material user-defined field. (Alphanumeric)-->
  <mat_udf3 /> (Optional)
        <!--material user-defined field. (Alphanumeric)-->
 </material>
 </materials>
</bcqci>
```

Batch Results

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <batchresults>
 <batchresult>
 <batchinfo>
  <bt_ticket /> (Required)
        <!--Ticket code that uniquely identifies this ticket at this plant (Alphanumeric)-->
  <bt_plantcode /> (Required)
        <!--User-defined alphanumeric code that uniquely identifies this plant (Alphanumeric)-->
  <bt_mixcode /> (Required)
        <!--User-defined alphanumeric code that uniquely identifies this mix at this plant
        (Alphanumeric)-->
  <bt_custcode /> (Required)
        <!--User-defined alphanumeric code that uniquely identifies this customer (Alphanumeric)--
        >
  <bt_custname /> (Optional)
        <!—Name of customer (Alphanumeric)-->
  <bt_jobcode /> (Optional)
        <!--User-defined alphanumeric code that uniquely identifies this job (Alphanumeric)-->
  <bt_jobname /> (Optional)
        <!—Name of job (Alphanumeric)-->
  <bt_loaddate /> (Required)
        <!--Load-out date (yyyy-mm-dd)-->
  <bt_loadtime /> (Required)
        <!--Load-out time (24 hour time, for example 15:00:00)-->
  <bt_loadunittype /> (Optional)
        <!--load unit type (Enumeration)-->
        <!--If not provided US should be assumed-->
  <bt_loadsize /> (Required)
        <!--load-out size (single truck) (Numeric)-->
  <bt_loadsizeunit /> (Optional)
        <!--load unit (Enumeration)-->
        <!--If not provided defaults are US=yd3, Metric=m3-->
  <bt_waterintruck /> (Optional)
        <!--water quantity in truck prior to load-out (Numeric)-->
```

<!--If not provided default is 0-->
<bt_waterintruckunit /> (Optional)
<!--water in truck unit (Enumeration)-->
<!--If not provided defaults are US=gallon, Metric=liter-->
<bt_watertrim /> (Optional)
<!--water removed from load-out that will be added onsite (Numeric)-->
<!--If not provided default is 0-->
<bt_watertrimunit /> (Optional)
<!--water trim unit (Enumeration)-->
<!--If not provided defaults are US=gallon, Metric=liter-->
<bt_batcher /> (Optional)
<!—name of person performing the batch (Alphanumeric)-->
<bt_truck /> (Optional)
<!--truck number or other identification (Alphanumeric)-->
<bt_driver /> (Optional)
<!--truck drivers name (Alphanumeric)-->
<batchinfo>
<batchcomponents>
<batchcomponent>
<bc_ticket /> (Required)
<!--User-defined code that matches bt_ticket (Alphanumeric)-->
<bc_matcode /> (Required)
<!--User-defined code that uniquely identifies this component in the mix (Alphanumeric)-->
<!--should match a material code at plant-->
<bc_mattype /> (Required)
<!--Material type (Enumeration)-->
<bc_design /> (Required)
<!--Batch material design quantity independent of load size (Numeric)-->
<bc_target /> (Required)
<!--Batch material target quantity for load size (Numeric)-->
<bc_actual /> (Required)
<!--Batch material actual quantity loaded (Numeric)-->
<bc_unit /> (Required)
<!--Batch material quantity units (Enumeration)-->
<bc_freemoisture /> (Optional)
<!--Aggregate free moisture % (Numeric)-->
<!--if provided then total moisture and absorption are not required→
<!--If not provided default is 0-->
<bc_totalmoisture /> (Optional - Required if free moisture not provided)
<!--Aggregate total moisture % (Numeric)-->
<!--necessary to compute free moisture→
<!--If not provided default is 0-->
<bc_absorption /> (Optional - Required if free moisture not provided or assumed 0)
<!--Aggregate absorption % (Numeric)-->
<!--necessary to compute free moisture→
<!--If not provided default is 0-->

```
 </batchcomponent>
 </batchcomponents>
 </batchresult>
 </batchresults>
</bcqci>
```

Batch Results Condensed

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <batchresults>
 <batchresult>
  <br_ticket /> (Required)
        <!--User-defined code that represents a ticket (Alphanumeric)-->
  <br_loaddate /> (Required)
        <!--Load-out date (yyyy-mm-dd)-->
  <br_loadtime /> (Required)
        <!--Load-out time (24 hour time, for example 15:00:00)-->
  <br_loadsize /> (Required)
        <!--load-out size (single truck) (Numeric)-->
  <br_loadsizeunit /> (Optional)
        <!--load unit (Enumeration)-->
        <!--If not provided defaults are US=yd3, Metric=m3-->
  <br_mixcode /> (Required)
        <!--User-defined alphanumeric code that uniquely identifies this mix at this plant
        (Alphanumeric)-->
  <br_matcode /> (Required)
        <!--User-defined code that uniquely identifies this component in the mix (Alphanumeric)-->
        <!--should match a material code at plant-->
  <br_mattype /> (Required)
        <!--Material type (Enumeration)-->
  <br_target /> (Required)
        <!--Batch material target quantity for load size (Numeric)-->
  <br_actual /> (Required)
        <!--Batch material actual quantity loaded (Numeric)-->
  <br_unit /> (Required)
        <!--Batch material quantity units (Enumeration)-->
</batchresult>
</batchresults>
</bcqci>
```

Customer

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<bcqci>
 <customers>
 <customer>
  <cust_plantcode /> (Optional)
        <!--User-defined code that uniquely identifies this plant (Alphanumeric)-->
  <cust_code /> (Required)
        <!--User-defined code that uniquely identifies this customer (Alphanumeric)-->
  <cust_name /> (Required)
        <!—Company name (Alphanumeric)-->
  <cust_first_name /> (Optional)
        <!--Customer first name (Alphanumeric)-->
  <cust_last_name /> (Optional)
        <!--Customer last name (Alphanumeric)-->
  <cust_address /> (Optional)
        <!--Customer address (Alphanumeric)-->
  <cust_city /> (Optional)
        <!--Customer city (Alphanumeric)-->
  <cust_state /> (Optional)
        <!--Customer state/province (Alphanumeric)-->
  <cust_zip /> (Optional)
        <!--Customer postal code (Alphanumeric)-->
  <cust_country /> (Optional)
        <!--Customer country (Alphanumeric)-->
  <cust_phone /> (Optional)
        <!--Customer phone (Alphanumeric)-->
  <cust_mobile /> (Optional)
        <!--Customer mobile (Alphanumeric)-->
  <cust_fax /> (Optional)
        <!--Customer fax (Alphanumeric)-->
  <cust_email /> (Optional)
        <!--Customer email (Alphanumeric)-->
  <cust_url /> (Optional)
        <!--Customer url/website (Alphanumeric)-->
  <cust_udf1 /> (Optional)
        <!--Customer user-defined field. (Alphanumeric)-->
  <cust_udf2 /> (Optional)
        <!--Customer user-defined field. (Alphanumeric)-->
  <cust_udf3 /> (Optional)
        <!--Customer user-defined field. (Alphanumeric)-->
 </customers>
 </customer>
</bcqci>
```

Projects

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<bcqci>
 <projects>
 <project>
  <cust_code /> (Required)
        <!--User-defined code that uniquely identifies the customer (Alphanumeric)-->
  <proj_code /> (Required)
        <!--User-defined code that uniquely identifies this project (Alphanumeric)-->
  <proj_name /> (Required)
        <!--Project name (Alphanumeric)-->
  <proj_address /> (Optional)
        <!—Project address (Alphanumeric)-->
  <proj_city /> (Optional)
        <!--Project city (Alphanumeric)-->
  <proj_state /> (Optional)
        <!--Project state/province (Alphanumeric)-->
  <proj_zip /> (Optional)
        <!--Project postal code (Alphanumeric)-->
  <proj_country /> (Optional)
        <!--Project country (Alphanumeric)-->
  <proj_desc /> (Optional)
        <!--Project description (Alphanumeric)-->
  <proj_location /> (Optional)
        <!--Project location (Alphanumeric)-->
  <proj_contractor /> (Optional)
        <!—Project contractor (Alphanumeric)-->
  <proj_contractid /> (Optional)
        <!—Project contract number (Alphanumeric)-->
  <proj_startdate /> (Optional)
        <!—Project Start Date  (yyyy-mm-dd)-->
  <proj_enddate /> (Optional)
        <!—Project End Date  (yyyy-mm-dd)-->
  <proj_udf1 /> (Optional)
        <!—Project user-defined field. (Alphanumeric)-->
  <proj_udf2 /> (Optional)
        <!--Project user-defined field. (Alphanumeric)-->
  <proj_udf3 /> (Optional)
        <!--Project user-defined field. (Alphanumeric)-->
 </projects>
 </project>
</bcqci>
```

Plants

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
  <plants>
        <plant>
```

```
    <plant_code /> (Required)
   <!--User-defined code that uniquely identifies a plant (Alphanumeric)-->
    <plant_name /> (Optional)
   <!--User-defined code that uniquely identifies a plant (Alphanumeric)-->
    <plant_maxloadsize /> (Optional)
   <!—Maximum load size for plant (Numeric)-->
    <plant_maxloadsizeunit /> (Optional)
   <!—Maximum load unit for plant (Enumeration)-->
   <!--If not provided cubic yards is assumed -->
    <plant_aggscalecapacity /> (Optional)
   <!—Aggregate scale capacity for plant (Numeric)-->
    <plant_cemscalecapacity /> (Optional)
   <!—Cement scale capacity for plant (Numeric)-->
    <plant_scalecapacityunit /> (Optional)
   <!—Plant scale capacities unit (Enumeration)-->
   <!--If not provided pounds is assumed -->
    <plant_timestamp /> (Optional)
   <!--If communicating computers are in different time zone or if the set time on the plant
   computer is accurate (Alphanumeric)-->
    <plant_software_mfg/> (Optional)
   <!--Name of software system manufacturer (Alphanumeric)-->
    <plant_software_model/> (Optional)
   <!--Name of software model/product line   (Alphanumeric)-->
    <plant_software_version /> (Optional)
   <!--Version of software (Alphanumeric)-->
   </plant>
  </plants>
</bcqci>
```

Plant List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <plantlist>
 <listitem>
   <plant_code />
   <plant_name />
 </listitem>
 </plantlist>
</bcqci>
```

Customer List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <customerlist>
```

```
 <listitem>
  <cust_code />
  <cust_name />
 </listitem>
 </customerlist>
</bcqci>
```

Project List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <projectlist>
 <listitem>
  <cust_code />
  <proj_code />
  <proj_name />
 </listitem>
 </projectlist>
</bcqci>
```

Material List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <materiallist>
 <listitem>
  <mat_plantcode />  <!—Required only if contains materials for multiple plants-->
  <mat_code />
  <mat_name />
  <mat_type />
 </listitem>
 </materiallist>
</bcqci>
```

Mix Design List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <mixdesignlist>
 <listitem>
  <mix_plantcode />  <!—Required only if contains mixes for multiple plants-->
  <mix_code />
  <mix_name />
 </listitem>
 </mixdesignlist>
</bcqci>
```

Batch Results List

```
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
 <batchresultlist>
 <listitem>
  <bt_ticket />
  <bt_loaddate />
  <bt_loadtime />
  <bt_loadsize /> <!--Optional-->
  <bt_mixcode />
 </listitem>
 </batchresultlist>
</bcqci>
```

Enumerations and Formats

All date formats are yyyy-mm-dd
All time formats are 24-hour time (15:00:00 for 3pm)

Culture
en = English (Default)
en-gb = English United Kingdom
fr = French
de = German
es = Spanish

Material Type
0 = Unspecified
1 = Coarse Aggregate
2 = Fine Aggregate
3 = Aggregate (undifferentiated)
4 = Cement
5 = Cement Alternative (fly ash, etc)
6 = Cement (undifferentiated)
7 = Admixture
8 = Water

Aggregate Weight Type
0 = Oven-Dry
1 = SSD

Dosage Type
0 = Per Cubic Yard/Cubic Meter
1 = Per Hundred Weight Total Cementitious
2 = Per Hundred Weight Cement
3 = Per Load

Unit Type
0 = US
1 = Metric

Unit
0 = None
1 = inches
2 = feet
5 = ounces
6 = pounds
7 = short tons
10 = fluid ounces

11 = gallons
15 = cubic feet
16 = cubic yard
20 = psi
25 = minutes
26 = hours
30 = percent

101 = millimeters
102 = centimeters
103 = meters
105 = grams
106 = kilograms
107 = metric tons
110 = milliliters
111 = liters
115 = cubic centimeters
116 = cubic meters
120 = MPa

Status

200 OK
201 Created
204 No Content

400 Bad Request
401 Unauthorized
404 Not Found, 10 = unknown/unsupported resource specifier
405 Method Not Allowed
409 Conflict, database error (usually a unique constraint on add/update or foreign key constraint on a delete)

500 Internal Server Error, unknown error (catch all)
501 Not Implemented, 7 = unsupported or unknown http method (HEAD, TRACE, etc.)
503 Service Unavailable

Notes

When receiving mix designs, the plant code should be used to verify that it is a valid plant; the mix code should be used to search for this mix at this plant; if the mix code exists then the mix and mix components should be updated/replaced; if the mix code does not exist then the mix and mix components should be created at the plant; if mix component material codes do not exist at the plant then plant materials should be created; plant materials typically should not be updated from mix components.

When receiving materials, the plant code should be used to verify that it is a valid plant; the mat code should be used to search for this material at this plant; if the mat code exists then the material should be updated/replaced; if the mat code does not exist then the material should be created at the plant.

When receiving customers, the plant code should be used to verify that it is a valid plant (if applicable); the customer code should be used to search for this customer at this plant; if the customer code exists then the customer should be updated/replaced; if the customer does not exist then the customer should be created.

- The XML definitions identify whether a field is alphanumeric, numeric, integer, or defined by an enumeration (which are either integer or alphanumeric).

- The XML definitions identify whether a field is required, optional, or has a requirement dependency.

- The XML definitions identify defaults when fields are not provided.

- Field tags can have a value, be included but left blank, or not included in the xml. These distinctions are important as not all systems store the same information. We will use mix component tag mc_order as an example. Quality systems may not store the rank or order in which mix components should be batched. As a result, we need a way to ensure that the batch software handles these different situations.

    - If the mc_order tag is provided and has a value then it should be assumed that these values should be replaced.
    - If the mc_order tag is provided and has no value then default values should be assumed or they should not be replaced.
    -If the mc_order tag is not provided then it should be assumed that the sending software has no knowledge of this tag. If the component did exist then these values should not be replaced. If the component did not exist then default values (possibly from the plant material) would be used.

We will consider cust_address as another example.

    - If the cust_address tag is provided and has a value then it should be assumed that these values should be replaced.

- If the cust_address tag is provided and has no value then it should be replaced as a blank value.
-If the cust_address tag is not provided then it should be assumed that the sending software has no knowledge of this tag. If the customer doesn't exist and is being created then cust_address would use the default value for this field, which is likely an empty string or null value. If the customer does exist and is being updated then the cust_address field should not be replaced.

XML Special Characters

XML predefines the following five entity references for special characters that would otherwise be interpreted as part of markup language:

Character Name	Entity Reference	Character Reference
Ampersand	&amp;	&
Left angle bracket	&lt;	<
Right angle bracket	&gt;	>
Straight quotation mark	&quot;	"
Apostrophe	&apos;	'

Therefore, all text fields that will be transmitted as xml need to have the special characters replaced with the entity reference. For example, a customer named B&B Contracting would become B'B Contracting. This can be accomplished using HTML encoding and decoding.

Pseudo-Webmethods

CustomerList (plantcode string) xml
Customer (plantcode string, custcode string) xml
Customer (customer xml) string

MaterialList (plantcode string) xml
Material (plantcode string, matcode string) xml
Material (material xml) string

MixDesignList (plantcode string) xml
MixDesign (plantcode string, mixcode string,) xml
MixDesign (mixdesign xml) string

BatchResultsList (plantcode string, ticket string) xml
BatchResultsList (plantcode string, startdate date; enddate date) xml
BatchResults(plantcode string, ticket string) xml
BatchResults(plantcode string, startdate date; enddate date) xml

Actual web methods may differ for each software depending on how web methods and security are implemented. When sending method parameters, such as when requesting a single material, URL encoding must be used. The following examples show proper URL encoding (UTF-8) for some example strings:

3/4 Rock = 3%2f4+Rock

D & M = D+%26+M

Project's = Project%27s

Date and Time Format

The date format is yyyy-mm-dd. The time format is 24-hour time such as hh:mm:ss. Date and time in the xml schemas have been defined as separate fields. However, some method calls, such as /batchresultlist/{plant_code}/{startdatetime}/{enddatetime}/{maxtickets} or /batchresults/{plant_code}/{datetime}/{datetime}, provide a means to send the date and time as follows: yyyy-mm-dd hh:mm:ss. For example: /batchresultlist/01/2015-01-01 10:15:20/2015-02-01 09:10:59/0

Security

Software System Dependent

Proposed RESTful API

Basics

An implementation of a REST Web service follows four basic design principles:

- Use HTTP methods explicitly.
- Be stateless.
- Expose directory structure-like URIs.
- Transfer XML, JavaScript Object Notation (JSON), or both.

Use HTTP methods explicitly

This basic REST design principle establishes a one-to-one mapping between create, read, update, and delete operations and HTTP methods:

- To create a resource on the server, and have the server return the primary key, use POST.
- To retrieve a resource, use GET.
- To create a resource with a provided primary key, change the state of a resource or to update it, use PUT.
- To remove or delete a resource, use DELETE.

Be stateless

A REST Web service application (or client) includes within the HTTP headers and body of a request all of the parameters, context, and data needed by the server-side component to generate a response.

Expose directory structure-like URIs

RESTful URLs or Clean URLs are a way to specify all necessary query parameters without the implementation details. For example the RESTful URL:
https://localhost/controlclient/api_v1/xml/customers/0101
specifies the resource “Customers” and a customer_code = “0101”.

Transfer XML, JavaScript Object Notation (JSON), or both

This has been covered extensively previously in this document, however, an example customer update would look like this. Note that for consistency the xml/json schema is the same regardless of whether it is a GET, PUT, or POST.

```
PUT /controlclient/api_v1/xml/customers/0101
HTTP/1.1
Host: localhost
Content-Type: application/xml
<?xml version="1.0" encoding="utf-8"?>
<bcqci>
    <customers>
        <customer>
                <cust_plantcode>301</cust_plantcode>
                <cust_code>0101</cust_code>
                <cust_name>Windsor's Concrete &amp;amp;Paving</cust_name>
```

```
            <cust_address>126 6th Street</cust_address>
            <cust_city>Windsor</cust_city>
            <cust_state>CO</cust_state>
            <cust_zip>80524</cust_zip>
            <cust_country>USA</cust_country>
            <cust_phone>970-674-1148</cust_phone>
            <cust_mobile></cust_mobile>
            <cust_fax/>
            <cust_email>info@stonemont.com</cust_email>
            <cust_url/>
        </customer>
    </customers>
</bcqci>
```

Proposed Interface Methods

The following table details the operations to be supported for this interface. Developing methods that include plant_code may depend on whether the system being called is a single-plant batching system or a multi-plant dispatch system. Additional methods may be required for specific batching or dispatch systems.

URI	Method	Resource	Operation
/connection/	GET	header	Returns status to validate connection
/plants/	GET	Plants	Retrieve full list of
/plants/{plant_code}	GET	Plants	Retrieve one
/mixdesigns/{plant_code}	GET	Mix Designs	Retrieve full list of mixes at plant
/mixdesigns/{plant_code}/{mix_code}	GET	Mix Designs	Retrieve one mix at plant
/mixdesigns/{plant_code}/{mix_code}	PUT	Mix Designs	Update or Create one
/mixdesigns/	POST	Mix Designs	Create, return Code
/materials/{plant_code}	GET	Materials	Retrieve full list of materials at plant
/materials/{plant_code}/{mat_code}	GET	Materials	Retrieve one material at plant
/materials/{plant_code}/{mat_code}	PUT	Materials	Update or Create one
/materials/	POST	Materials	Create, return Code
/batchresults/{plant_code}/{date}	GET	Batch Results	Retrieve full list at plant for single date
/batchresults/{plant_code} /{datetime}/{datetime}	GET	Batch Results	Retrieve full list at plant for date range
/batchresultsmix/{plant_code}/{mix_code} /{date}	GET	Batch Results	Retrieve full list at plant for single date
/batchresultsmix/{plant_code}/{mix_code} /{datetime}/{datetime}	GET	Batch Results	Retrieve full list at plant for date range
/batchresultsticket/{plant_code} /{ticket_num}	GET	Batch Results	Retrieve one at plant
/batchresultscondensed/{plant_code} /{startdatetime}/{enddatetime}/{minloadsize} /{maxtickets}	GET	Batch Results Condensed	Retrieve for plant and date range. If maxtickets = 0 then return all tickets otherwise return number of most recent tickets.

/batchresultscondensed/{plant_code} /{mix_code}/{startdatetime}/{enddatetime} /{minloadsize}/{maxtickets}	GET	Batch Results Condensed	Retrieve specific mix at plant for date range. If maxtickets = 0 then return all tickets otherwise return number of most recent tickets.
/customers/	GET	Customers	Retrieve all customers
/customerssince/{date}	GET	Customers	Retrieve customers created or modified since date
/customers/{cust_code}	GET	Customers	Retrieve one customer
/customers/{cust_code}	PUT	Customers	Update or Create one
/customers/	POST	Customers	Create, return Code
/projects/{cust_code}	GET	Projects	Retrieve all projects for a customer
/projectssince/{date}	GET	Projects	Retrieve projects created or modified since date
/projects/{cust_code}/{proj_code}	GET	Projects	Retrieve one
/projects/{cust_code}/{proj_code}	PUT	Projects	Update or Create one
/projects/	POST	Projects	Create, return Code
LISTS			
/mixdesignlist/{plant_code}	GET	Mix Design List	Returns mix design list for all mixes at plant
/mixdesignlist/{plant_code}/{mix_code}	GET	Mix Design List	Returns mix design list for single mix at plant.
/materiallist/{plant_code}	GET	Material List	Returns material list for plant
/materiallist/{plant_code}/{mat_code}	GET	Material List	Returns material list for single material at plant.
/batchresultlist/{plant_code}/{date} /{maxtickets}	GET	Batch Results List	Returns batch results list for a single date at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultlist/{plant_code} /{startdatetime}/{enddatetime} /{maxtickets}	GET	Batch Results List	Returns batch results list between dates at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.

/batchresultlist/{plant_code} /{startdatetime}/{enddatetime} /{maxtickets}	GET	Batch Results List	Returns batch results list between dates/times at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultmixlist/{plant_code} /{mix_code}/{date}/{maxtickets}	GET	Batch Results List	Returns batch results list for a single date and single mix at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultmixlist/{plant_code} /{mix_code} /{startdatetime}/{enddatetime} /{maxtickets}	GET	Batch Results List	Returns batch results list between dates for a single mix at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultcustlist/{plant_code}/{cust_code} /{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Results List	Returns batch results list between dates/times for a specific customer at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultcustmixlist/{plant_code}/{cust_code} /{mix_code}/{startdatetime}/{enddatetime} /{maxtickets}	Get	Batch Results List	Returns batch results list between dates/times for a specific customer and mix at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultcustjoblist/{plant_code}/{cust_code} /{job_code}/{startdatetime}/{enddatetime} /{maxtickets}	Get	Batch Results List	Returns batch results list between dates/times for a specific customer and job at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/batchresultcustjobmixlist/{plant_code}/{cust_code} /{job_code}/{mix_code}/{startdatetime} /{enddatetime}/{maxtickets}	Get		Returns batch results list between dates/times for a specific customer, job, and mix at a plant. If maxtickets = 0 then return all otherwise return number of oldest tickets.
/plantlist/	GET	Plant List	Retrieve full list of plants
/customerlist/	GET	Customer List	Returns customer list for all customers
/customerssincelist/{date}	GET	Customer List	Returns customer list for all customers created or modified since date

/customercontainslist/{search}	GET	Customer List	Returns customer list for all customer names that include/contains search
/projectlist/{cust_code}	GET	Project List	Returns project list for a customer
/projectssincelist/{cust_code}/{date}	GET	Project List	Returns project list for a customer created or modified since date

File Transfer Folders

Ideally the BCQI is to be implemented via a RESTful web service. However, if it is not possible to implement a RESTful web service a file transfer approach can be used instead.

Master Folder
\\ComputerName\SoftwareName\PlantCode\

SoftwareName is that of the batch software. Export refers to the batch software exporting files to this location. Import refers to the batch software importing files from this location.

Fixed Export Sub Folder Names
ExportMixDesigns
ExportMaterials
ExportCustomers
ExportBatchResults

Fixed Import Sub Folders Names
ImportMixDesigns
ImportMaterials
ImportCustomers

APPENDIX 1

Kronware Concrete Rhine ABS

The Concrete Rhino ABS BCQCI exposed resources include:

- Customers
- Materials
- Mix Designs
- Batch Results
- Plants

The following table details the methods supported for this interface.

URI	Method	Resource	Operation
/customers/	GET	Customers	Retrieve full list of
/customers/{cust_code}	GET	Customers	Retrieve one
/customers/{cust_code}	PUT	Customers	Update or Create one
/customers/	POST	Customers	Create, return Code
/materials/	GET	Materials	Retrieve full list of
/materials/{material_code}	GET	Materials	Retrieve one
/materials/{material_code}	PUT	Materials	Update or Create one
/materials/	POST	Materials	Create, return Code
/mixdesigns/	GET	Mix Designs	Retrieve full list of
/mixdesigns/{mix_code}	GET	Mix Designs	Retrieve one
/mixdesigns/{mix_code}	PUT	Mix Designs	Update or Create one
/mixdesigns/	POST	Mix Designs	Create, return Code
/batchresults/	GET	Batch Results	Retrieve full list of
/batchresults/{ticket_num}	GET	Batch Results	Retrieve one
/batchresults/{date}/{date}	GET	Batch Results	Retrieve full list by date range
/batchresultlist/	GET	Batch Results List	Retrieve listing of all
/batchresultlist/{date}	GET	Batch Results List	Retrieve summary list by single date
/batchresultlist/{date}/{date}	GET	Batch Results List	Retrieve summary list by date range
/mixdesignlist/	GET	Mix Design List	Retrieve listing of all

/customerlist/	GET	Customer List	Retrieve listing of all
/materiallist/	GET	Material List	Retrieve listing of all
/plants/	GET	Plants	Retrieve full list of
/plants/{plant_code}	GET	Plants	Retrieve one

APPENDIX 2

GivenHansco Keystone Dispatch

The GivenHansco Keystone Dispatch BCQCI exposed resources include:

- Customers
- Projects
- Materials
- Mix Designs
- Batch Results
- Plants

The following table details the methods supported for this interface.

URI	Method	Resource
/batchresults/{plant_code}/{startdatetime}/{enddatetime}	Get	Batch Results
/batchresults/{plant_code}/{date}	Get	Batch Results
/batchresultsmix/{plant_code}/{mix_code}/{startdatetime}/{enddatetime}	Get	Batch Results
/batchresultsmix/{plant_code}/{mix_code}/{date}	Get	Batch Results
/batchresultsticket/{plant_code}/{ticket}	Get	Batch Results
/batchresultscondensed/{plant_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Results Condensed
/batchresultscondensed/{plant_code}/{mix_code}/{begindatetime}/{enddatetime}/{maxtickets}	Get	Batch Results Condensed
/batchresultlist/{plant_code}/{startdatetime}/{maxtickets}	Get	Batch Result List
/batchresultlist/{plant_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Result List
/batchresultmixList/{plant_code}/{mix_code}/{startdatetime}/{maxtickets}	Get	Batch Result List
/batchresultmixlist/{plant_code}/{mix_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Result List
/connection	Get	Header
/customers	Get	Customers
/customers/{cust_code}	Get	Customers
/customerssince/{sincedate}	Get	Customers
/customerlist	Get	Customer List

/materiallist	Get	Material List
/materiallist/{plant_code}	Get	Material List
/materiallist/{plant_code}/{mat_code}	Get	Material List
/materials	Get	Materials
/materials/{plant_code}	Get	Materials
/materials/{plant_code}/{mat_code}	Get	Materials
/mixdesigns	Get	Mix Designs
/mixdesigns/{plant_code}	Get	Mix Designs
/mixdesigns/{plant_code}/{mat_code}	Get/Put	Mix Designs
/mixdesignlist	Get	Mix Design List
/mixdesignlist/{plant_code}	Get	Mix Design List
/mixdesignlist/{plant_code}/{mat_code}	Get	Mix Design List
/plants	Get	Plants
/plants/{plant_code}	Get	Plants
/projects/{cust_code}	Get	Projects
/projects/{cust_code}/{proj_code}	Get	Projects
/projectssince/{sincedate}	Get	Projects
/projectlist/{cust_code}	Get	Project List
/projectlist	Get	Project List

APPENDIX 3

MPAQ Batch

The MPAQ Batch BCQCI exposed resources include:

- Materials
- Mix Designs
- Batch Results
- Plants

The following table details the methods supported for this interface.

URI	Method	Resource
/batchresults/{plant_code}/{startdatetime}/{enddatetime}	Get	Batch Results
/batchresults/{plant_code}/{date}	Get	Batch Results
/batchresultsticket/{plant_code}/{ticket}	Get	Batch Results
/batchresultscondensed/{plant_code}/{startdatetime}/{enddatetime}/{minloadsize} /{maxtickets}	Get	Batch Results Condensed
/batchresultscondensed/{plant_code}/{mix_code}/{begindatetime}/{enddatetime}/{minloadsize} /{maxtickets}	Get	Batch Results Condensed
/batchresultlist/{plant_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Result List
/batchresultmixlist/{plant_code}/{mix_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Result List
/batchresultcustlist/{plant_code}/{cust_code}/{startdatetime}/{enddatetime}/{maxtickets}	Get	Batch Result List
/batchresultcustmixlist/{plant_code}/{cust_code}/{mix_code}/{startdatetime}/{enddatetime} /{maxtickets}	Get	Batch Result List
/batchresultcustjoblist/{plant_code}/{cust_code}/{job_code}/{startdatetime}/{enddatetime} /{maxtickets}	Get	Batch Result List
/batchresultcustjobmixlist/{plant_code}/{cust_code}/{job_code}/{mix_code}/{startdatetime} /{enddatetime}/{maxtickets}	Get	Batch Result List
/connection	Get	Header
/materiallist/{plant_code}	Get	Material List
/materials/{plant_code}	Get	Materials
/materials/{plant_code}/{mat_code}	Get/Put	Materials
/mixdesigns/{plant_code}	Get	Mix Designs

/mixdesigns/{plant_code}/{mat_code}	Get/Put	Mix Designs
/mixdesignlist/{plant_code}	Get	Mix Design List
/plantlist	Get	Plants
/plants/{plant_code}	Get	Plants